



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/872,085	06/01/2001	Joseph R. Hunt	10001579-1	3829

7590 02/24/2005
HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400

EXAMINER

RAMPURIA, SATISH

ART UNIT	PAPER NUMBER
----------	--------------

2124

DATE MAILED: 02/24/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.		Applicant(s)	
	09/872,085		HUNT ET AL.	
	Examiner		Art Unit	
	Satish S. Rampuria		2124	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 03 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 20 September 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-33 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-33 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

Response to Amendment

1. This action is in response to the amendment received on Sep 20, 2004.
2. The objection use of trademarks is withdrawn in view of applicant's amendment.
3. The rejection under 35 U.S.C. §112 to claim 1 is withdrawn in view of applicant's amendment.
4. The rejections under 35 U.S.C. §101 to claims 1 and 22 is withdrawn in view of applicant's amendment.
5. Claims canceled by the applicant -- **None**.
6. Claims amended by the applicant -- **1 and 22**.
7. Claims pending in the application -- **1-33**.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1-6, 12-14, 16, 19, 20, 22-29, and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 5,943,497 to Bohrer et al., hereinafter called Bohrer, in view of US Patent No. 6,438,743 to Boehm et al., hereinafter called Boehm.

Per claims 1, 4, 12, and 19:

Bohrer disclose:

Application/Control Number: 09/872,085

Art Unit: 2124

- A method for run-time configurable caching of factory objects (col. 2, lines 24-27 “an object-oriented mechanism is disclosed that allows new configuration data to replace existing configuration data within an existing object-oriented program”), comprising:
 - launching an object-oriented application of a computer. It is inherent to launch the object-oriented application to work with objects.
 - assigning one or more cache objects of the plurality of cache objects to one or more cacheable factory objects of the plurality of cacheable factory objects (col. 9, lines 46-50 “the NamingService object 540 uses a cache memory to quickly access the most recently accessed configuration data objects. If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object”);

Bohrer does not explicitly disclose creating and initializing a plurality of cacheable factory objects, said plurality of cacheable factory objects residing in a memory of the computer; creating and initializing a plurality of cache objects, said plurality of cacheable factory objects residing in a memory of the computer; manipulating one or more objects contained in the plurality of cache objects.

However, Boehm discloses in an analogous computer system creating and initializing a plurality of cacheable factory objects (col. 3, lines 21-22 “present invention supports the creation and caching of identification files”); creating and initializing a plurality of cache objects (col. 3, lines 21-22 “present invention supports the creation and caching of identification files”); manipulating one or more objects contained in the plurality of cache

Application/Control Number: 09/872,085

Art Unit: 2124

objects (col. 6, lines 61-62 "manipulate and maintain the network caches outside the build process").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of creating and initializing a plurality of cache objects and manipulating objects to maintain cache as taught by Boehm into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because of one of ordinary skill in the art would be motivated to use the cache objects and manipulating to optimize the development environment as suggested by Boehm (col. 2, lines 14-32).

Per claims 2 and 13:

The rejection of claim 1 is incorporated, and further, Bohrer does not explicitly disclose the plurality of cache objects contain a plurality of methods to add an object to a cache, remove an object from the cache, and find an object in the cache.

However, Boehm discloses in an analogous computer system the plurality of cache objects contain a plurality of methods to add an object to a cache, remove an object from the cache, and find an object in the cache (col. 5, lines 56-58 "building a software system using cacheable items... includes a build list creation step..., a network cache load..., a build program... and a network cache maintenance and update (add an object, remove an object, and find an object) step")

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of updating the cache objects as taught by Boehm

Application/Control Number: 09/872,085

Art Unit: 2124

into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because of one of ordinary skill in the art would be motivated to use the cache objects and manipulating to optimize the development environment as suggested by Boehm (col. 2, lines 14-32).

Per claim 3:

The rejection of claim 1 is incorporated, and further, Bohrer disclose:

- wherein the plurality of cacheable factory objects contain a plurality of methods to get an object from a cache, and to couple a cache object to a cacheable factory object (col. 9, lines 47-50 "If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object")

Per claim 5:

The rejection of claim 4 is incorporated, and further, Bohrer does not explicitly disclose the plurality of factory objects contain a plurality of methods to create an object, obtain an object identifier, and get a database connection object.

However, Boehm discloses in an analogous computer system the plurality of factory objects contain a plurality of methods to create an object, obtain an object identifier, and get a database connection object (col. 3, lines 1-2 "identification and searching the network caches and creating the links to cached source and objects files").

Application/Control Number: 09/872,085

Art Unit: 2124

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of create an object and obtain an identifier linking to an object in database as taught by Boehm into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because of one of ordinary skill in the art would be motivated to create cache objects and obtain an identifier linking to an object in database to optimize the development environment as suggested by Boehm (col. 2, lines 14-32).

Per claim 6:

The rejection of claim 4 is incorporated, and further, Bohrer does not explicitly disclose:

- the plurality of factory objects and the plurality of cache objects derive from a common base object (col. 9, lines 46-50 "NamingService object 540 uses a cache memory to quickly access the most recently accessed configuration data objects. If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object")

Per claims 14 and 16:

The rejection of claim 13 is incorporated, and further, Bohrer disclose:

- the application sending a message to a cacheable factory object of the plurality of cacheable factory objects to add the one or more objects to the cache object coupled to the cacheable factory object (col. 9, lines 47-50 "If a factory requests a different

configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object”)

- the cacheable factory object receiving the message and sending a message to the cache object to add the one or more objects to the cache coupled to the cache object (col. 9, lines 47-50 “If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object”)

Per claim 20:

The rejection of claim 13 is incorporated, and further, Bohrer disclose:

- the application sending a message to a cacheable factory object of the plurality of cacheable factory objects to locate the one or more objects located in a cache object coupled to the cacheable factory object (col. 9, lines 47-50 “If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object”);
- the cacheable factory object sending a message to the cache object determining whether any of the one or more objects are contained in the cache object (col. 9, lines 47-50 “If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object” and col. 7, lines 62-65 “The factory would recognize the container ID and transfer control to the persistent container object”);

Art Unit: 2124

Bohrer does not explicitly disclose if able to locate the one or more objects, the cache object returning any of the one or more objects contained in the cache object; and if unable to locate the one or more objects, the cacheable factory object accessing the one or more objects from a database, and adding the one or more objects to the cache object.

However, Boehm discloses in an analogous computer system if able to locate the one or more objects, the cache object returning any of the one or more objects contained in the cache object (col. 9, lines 16-23 “if there is a copy of a source file stored in the source file cache... the source file handler creates a link... build list source file name to cache link name and cache address of... source file... places... link ... in the user directory structure”); and if unable to locate the one or more objects, the cacheable factory object accessing the one or more objects from a database, and adding the one or more objects to the cache object (col. 9, lines 24-33 “if the source file handler does not find a file having the cache link name in the network source file cache... source file handler checks out a copy of the source file from... library... renames...copies the renamed file in to the proper network cache... ”)

The feature of locating cache objects would be obvious for the reasons set forth in the rejection of claim 1.

Per claims 22 and 27:

Bohrer disclose:

- one or more cacheable factory objects residing in a memory of the computer, derived from the one or more factory objects (col. 9, lines 45-47 “NamingService object 540

Art Unit: 2124

uses a cache memory to quickly access the most recently accessed configuration data objects”); and

- one or more factory objects, coupled to an object-oriented application. It is inherent to have objects in object-oriented application to work with objects.

Bohrer does not explicitly disclose one or more cache objects residing in the memory of the computer, coupled to the cacheable factory objects.

However, Boehm discloses in an analogous computer system the one or more cache objects residing in the memory of the computer, coupled to the cacheable factory objects (col. 4, lines 41-42 “When searching the cached to locate... object file to create a link”).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of coupling the cache objects by creating a link to objects as taught by Boehm into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because of one of ordinary skill in the art would be motivated to couple the objects to optimize the development environment as suggested by Boehm (col. 2, lines 14-32).

Per claim 23:

The rejection of claim 22 is incorporated, and further, Bohrer does not explicitly disclose the object-oriented application interacts with the plurality of cacheable factory objects in order to manipulate one or more objects contained in the plurality of cache objects.

Art Unit: 2124

However, Boehm discloses in an analogous computer system manipulating one or more objects contained in the plurality of cache objects (col. 6, lines 61-62 “manipulate and maintain the network caches outside the build process”).

The feature of coupling cache objects would be obvious for the reasons set forth in the rejection of claim 22.

Per claim 24:

The rejection of claim 22 is incorporated, and further, Bohrer disclose:

- a plurality of objects contained in the one or more cache objects can be uniquely identified (col. 7, lines 49-50 “a container ID, which is a unique string indicating the types of support”)

Per claim 25:

The rejection of claim 22 is incorporated, and further, Bohrer does not explicitly disclose the plurality of cache objects contain a plurality of methods to add an object to a cache, remove an object from the cache, and find an object in the cache.

However, Boehm discloses in an analogous computer system the plurality of cache objects contain a plurality of methods to add an object to a cache, remove an object from the cache, and find an object in the cache (col. 5, lines 56-58 “building a software system using cacheable items... includes a build list creation step..., a network cache load..., a build program... and a network cache maintenance and update (add an object, remove an object, and find an object) step”)

Art Unit: 2124

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of updating the cache objects as taught by Boehm into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because of one of ordinary skill in the art would be motivated to use the cache objects and manipulating to optimize the development environment as suggested by Boehm (col. 2, lines 14-32).

Per claim 26:

The rejection of claim 22 is incorporated, and further, Bohrer disclose:

- wherein the plurality of cacheable factory objects contain a plurality of methods to get an object from a cache, and to couple a cache object to a cacheable factory object (col. 9, lines 47-50 “If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object”)

Per claim 28:

The rejection of claim 22 is incorporated, and further, Bohrer does not explicitly disclose the plurality of factory objects contain a plurality of methods to create an object, obtain an object identifier, and get a database connection object.

However, Boehm discloses in an analogous computer system the plurality of factory objects contain a plurality of methods to create an object, obtain an object identifier, and get a

Art Unit: 2124

database connection object (col. 3, lines 1-2 “identification and searching the network caches and creating the links to cached source and objects files”).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of create an object and obtain an identifier linking to an object in database as taught by Boehm into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because of one of ordinary skill in the art would be motivated to create cache objects and obtain an identifier linking to an object in database to optimize the development environment as suggested by Boehm (col. 2, lines 14-32).

Per claims 29 and 33:

The rejection of claim 4 is incorporated, and further, Bohrer does not explicitly disclose:

- the plurality of factory objects and the plurality of cache objects derive from a common base object (col. 9, lines 46-50 “NamingService object 540 uses a cache memory to quickly access the most recently accessed configuration data objects. If a factory requests a different configuration data not found in the cache, the NamingService object may then retrieve the needed configuration data from the ConfigurationData object”)

3. Claims 7-11, 15, 17, 18, and 30-32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bohrer in view of US Patent No. 6,446,188 to Henderson et al., hereinafter called Henderson.

Per claims 7, 8, 30, and 31:

Art Unit: 2124

The rejection of claim 1 is incorporated, and further, Bohrer does not explicitly disclose the plurality of cache statistics objects contain a plurality of methods to determine the number of cache accesses, the number of times a cache access returned an empty result, the size of a cache, and a reset command.

However, Henderson discloses in an analogous computer system the plurality of cache statistics objects contain a plurality of methods to determine the number of cache accesses (col. 3, lines 11-14 “the system comprises an object cache for caching frequently accessed memory”), the number of times a cache access returned an empty result, the size of a cache, and a reset command (col. 3, lines 7-10 “A system for dynamic memory management maps a sparsely populated virtual address space of memory objects to a more densely populated physical address space of fixed size memory elements for use by a host processor” and col. 8, lines 13-17 “Management registers... provide... data for the DMC... registers contain information...address translation module... the management module 404... contain results of host processor commands”).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of keeping track of cache as taught by Henderson into the method of caching of components of factory objects as taught by Bohrer. The modification would be obvious because one of ordinary skill in the art would be motivated to keep track of cache to manage systems require memory resources as suggested by Henderson (col. 2, lines 42-65).

Per claims 9 and 32:

Art Unit: 2124

The rejection of claim 7 is incorporated, and further, Bohrer does not explicitly disclose the plurality of cache configuration objects contain a plurality of methods to empty a cache, set and get a maximum cache size, and set and get the cache type.

However, Henderson discloses in an analogous computer system the plurality of cache configuration objects contain a plurality of methods to empty a cache (col. 8, lines 21-22 “the number of free entries in the management table cache”) set and get a maximum cache size (col. 8, lines 19-20 “The permanent registers 512 contain information such as the maximum size of a memory object”), and set and get the cache type (col. 8, lines 23-24 “a pointer to the next free entry in the management table cache”).

The feature of accessing cache would be obvious for the reasons set forth in the rejection of claims 7 and 8, respectively.

Per claims 10 and 17:

The rejection of claims 1 and 16 is incorporated, respectively, and further, neither Bohrer nor Boehm explicitly disclose setting a cache type for each cache object of the plurality of cache objects; and setting a maximum size for the number of objects contained in each cache object of the plurality of cache.

However, Henderson discloses in an analogous computer system setting a cache type for each cache object of the plurality of cache objects (col. 8, lines 41-43 “The initialize process sets the DMC-(Dynamic Memory Cache) and associated private memory to a known state” and col. 7, lines 35-36 “The object cache 406 provides a fast local memory used to store frequently accessed memory element data”); and setting a maximum size for the number of objects

Art Unit: 2124

contained in each cache object of the plurality of cache objects (col. 7, lines 63-65 “FIG. 5 also shows an example of three dynamically allocated memory objects of varying size added after DMC initialization”).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of setting cache type and setting a maximum size for the number of object as taught by Henderson into the method of caching of components of factory objects as taught by the combination system of Bohrer and Boehm. The modification would be obvious because of one of ordinary skill in the art would be motivated to keep track of cache to manage systems require memory resources as suggested by Henderson (col. 2, lines 42-65).

Per claims 11, 15, and 18:

The rejection of claim 10 is incorporated, and further, neither Bohrer nor Boehm explicitly disclose the cache type for each cache object of the plurality of cache objects determines how each object in the cache is removed from the cache.

However, Henderson discloses in an analogous computer system the cache type for each cache object of the plurality of cache objects determines how each object in the cache is removed from the cache (col. 8, lines 45-48 “The release process verifies that the dynamic memory release is valid and frees resources, removes memory objects, and updates status”).

The feature of accessing cache would be obvious for the reasons set forth in the rejection of claim 10.

Response to Arguments

8. Applicant's arguments with respect to claims have been considered but they are not persuasive.

In the remarks, the applicant has argued that:

- For claim 1, Bohrer does not teach, suggest, or anticipate of run-time configurable caching of factory objects (or component factories) and an assignment between cache objects and cacheable factory objects. Reference Boehm is outside the field of the invention and has no relevance to show said software application manages it's cache using objects or otherwise.
- For claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33 the combination of Bohrer and Boehm is improper because as mentioned in the response to the claim 1 rejection, reliance upon the Boehm reference in combination with the Bohrer as a ground of rejection is defective.
- For claims 22 and 27, the combination of Bohrer and Boehm does not the entirety of the claims. Bohrer does not teach or suggest of cacheable factory objects derived from one or more factory objects. Boehm does not teach or suggest how said software application manages it's cache using objects or otherwise.
- For claims 7-11, 15, 17, 18, and 30-32 the reference Henderson is outside of the field of the invention, having nothing to do with object oriented application.

Examiner's response:

- Regarding claim 1, Applicant appears to argue that the Bohrer and Boehm references are non-analogous. Bohrer system disclose that “allows new configuration data to replace existing configuration data within an existing object-oriented program... modified class is used to create the object”, see col. 2, lines 25-36 also, fig 5 and 6 and related discussion. Boehm system discloses creation and caching of a file corresponding to object file in cache “building a software system in a software development environment where multiple workstations have access to one or more network caches”, col. 2 and 3, lines 35-67 and 1-31. Therefore, reference Boehm is analogous in within the field of the invention. Further, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Applicant only makes general allegations and does not point out any errors in the rejection. The rejection clearly points out where Bohrer and Boehm teach the claimed features and why it would have been obvious to combine their teachings. Therefore, the rejection is proper and maintained herein.
- Regarding claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33, Applicant appears to argue that the rejection is improper because as mentioned in the response to the claim 1 rejection, reliance upon the Boehm reference in combination with the Bohrer as a ground of rejection is defective. See response to claim 1 above. Further, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of

references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Applicant only makes general allegations and does not point out any errors in the rejection. The rejection clearly points out where Bohrer and Boehm teach the claimed features and why it would have been obvious to combine their teachings. Therefore, the rejection is proper and maintained herein.

- Regarding claims 22 and 27, Bohrer does teach cacheable factory objects. Bohrer system disclose that “allows new configuration data to replace existing configuration data within an existing object-oriented program... modified class is used to create the object”, see col. 2, lines 25-36 also, fig 5 and 6 and related discussion. Therefore, reference Boehm is analogous and in within the field of the invention. It is noted that the rejection clearly points out where Bohrer and Boehm teach the claimed features and why it would have been obvious to combine their teachings. Applicant only makes general allegations and does not point out any errors in the rejection. Rather, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Therefore, the rejection is proper and maintained herein.
- Regarding claims 7-11, 15, 17, 18, and 30-32, Applicant appears to argue that Henderson is outside of the field of the invention, having nothing to do with object oriented application. Henderson discloses dynamic memory mapping where system comprises an

Art Unit: 2124

object cache for caching frequently accessed memory. Henderson does not disclose Object Oriented applications however, management of caching memory which is done in the computer art can very well be implemented in an Object Oriented application. Therefore, reference Henderson is analogous and in within the field of the invention. Applicant only makes general allegations and does not point out any errors in the rejection. It is noted that the rejection clearly points out where Bohrer and Henderson teach the claimed features and why it would have been obvious to combine their teachings. Further, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Therefore, the rejection is proper and maintained herein.

Conclusion

1. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).


A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

Application/Control Number: 09/872,085

Art Unit: 2124

however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Satish S. Rampuria
Patent Examiner
Art Unit 2124
02/22/2005


KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100